JOINT INTEROPERABILITY & ENGINEERING ORGANIZATION

CENTER FOR SOFTWARE

Management Plan MP

20 January 1995

SOFTWARE DEVELOPMENT PLAN (SDP)

FOR THE

AIRFIELDS SYSTEM

Version 2.0.1
CM Number: LL-521-03-02

(D R A F T)

SUBMITTED BY: **APPROVED BY:**

JAMES MOODY   **SAMUEL PUCCIARELLI**
Chief, General Applications   **Chief, Software**
** Division**   **Development Department**

Copies of this document may be obtained from:

The Director CFSW
Attn: Code JEXAG
5600 Columbia Pike
Falls Church, VA 22041

ACKNOWLEDGEMENT

CONTENTS

FIGURES

ABSTRACT

This SDP will be used by the Government to monitor the procedures and management efforts of the personnel performing the technical responsibilities involved in the development of the Airfields re-engineering effort under the Ada 95 Early Adoption Program. This SDP is developed in accordance with DoD-STD-2167A, Defense System Software Development, and the Joint Data Systems Support Center (JDSSC) Procedures Manual PM 2-90, dated 1 September 1990.

The format and content of this document follow the guidelines in JDSSC Procedures Manual PM 1-90, Documentation Standards and Publications Style Manual, and JDSSC Procedures Manual PM 2-90.

SECTION 1.   SCOPE

1.1   Identification

The Airfields system provides the Worldwide Military Command and
Control System (WWMCCS) community with a wide range of data about
free world airfields. The data is supplied by the Defense Mapping
Agency Aerospace Center (DMAAC) and is updated monthly.

This document addresses a plan to re-engineer/re-host the Airfields
application that currently exists as a WWMCCS COBOL application.
It will be re-engineered under the Ada 95 Early Adoption Program
using Ada 95 as the programming language, ORACLE Data Base
Management System (DBMS) Version 7, and a Windows/Motif compliant
Graphical User Interface (GUI), Screen Machine, development tool.
This plan deals with the issues and activities associated with the
transition from COBOL to Ada 95 and addresses those issues and
activities associated with traditional applications development.

1.2   Project Overview

The following paragraphs present brief management and system
overviews for Airfields.

1.2.1   Management Overview

Project management efforts consist of supervising the development
staff, providing the software development environment, ensuring
deadlines and milestones are met, and ensuring the project stays
within budget.

Table 1-1 depicts the personnel assigned to the project, their
primary responsibilities, and their commitment to the overall
transition effort as well as their specific time commitment to
Airfields. Figure 1-1 represents a flow diagram of the team
starting with the team's leader.

1.2.2   System Overview

Airfields will be developed in a Unix environment on a Sun Solaris
platform with report and retrieval manipulations accomplished using
ORACLE Version 7 as the database management system. Screen
Machine, a Windows/Motif compliant Graphical User Interface (GUI)
development tool will be utilized. The Airfields system will
comply with the GCCS Integration Standards.

1.3   Document Overview

The purpose of this SDP is to describe the plan for the re-
engineering/re-hosting of the Airfields system. It describes the

resources, organizational structure, software development standards and procedures, software configuration management procedures, and software quality assurance procedures to be used in this development effort. Development of the SDP is based on the Military Standard Defense System Software Development document, DoD-STD-2167A and the Standards and Procedures for Software Projects manual, PM 2-90.

1.4   References

   a.   Department of Defense, Military Standard Defense System Software Development , DoD-STD-2167A, 29 Feb 1988

   b.   Joint Data Systems Support Center (JDSSC), Documentation Standards and Publications Style Manual , PM 1-90, 1 August 1990

   c.   Joint Data Systems Support Center (JDSSC), Standards and Procedures for Software Projects , PM 2-90, 15 May 1988

   d.   Microsoft's ODBC 2.0 Programmer's Reference and SDK Guide for Microsoft Windows and Windows NT

   e.   Defense Mapping Agency Aerospace Center (DMAAC) Mapping and Charting Department Air Facilities System Input Instructions for the Automated Air Facilities Information File (AAFIF) (U), Draft , dated 1 May 1991

   f.   DMAAC Automated Air Facilities Information File (AAFIF) to ADM Cross-Reference Map , dated 15 December 1994

   g.   Defense Mapping Products Specifications for the AAFIF, First Edition (Draft) , dated October 1987

   h.   Defense Mapping Products Specifications for the AAFIF, Second Edition (Draft) , dated June 1996

   i.   Department of Defense/DISA/DSSO/JNSL Airfields User Instructions, Version 2.0 , dated 12 May 1990

   j.   International Organization for Standardization, International Electrotechnical Committee, Information Technology Programming Languages Their Environments and System Software Interfaces, Ada 9X Quality and Style Guidelines for Professional Programmers (Draft Baseline Version) , SPC-94093-CMC, Version 00.01.00, dated February 1995

   k.   International Organization for Standardization, International Electrotechnical Committee, Information

Technology Programming Languages Their Environments and
System Software Interfaces, Ada 9X Reference Manual
(Draft), Version 5.0, dated 1 June 1994

l.   International Organization for Standardization,
        International Electrotechnical Committee, Information
        Technology Programming Languages Their Environments and
        System Software Interfaces, Ada 9X Rationale (Draft),
        Version 5.0, dated 8 June 1994

1.5   Terms and Abbreviations

All terms and abbreviations are defined the first time they are
used in the document. A list for reference can be found in
Appendix A, Page A-1.

Assignment	Personnel	Commitment	
		Overall	To AIRFIELDS
Project Management	Velma Blue (Project Lead)	Full Time	100%
	Patrick Kelly	Full Time	25%
	Patricia Lewis	Part Time	25%
Technical Leads	Russell Smith	Full Time	50%
	Angela Arrington	Part Time	50%
Database Support	Rob Vietmeyer	Full Time	100%
	Russell Smith	Full Time	50%
	Fehmida Moorad	1/2 Time	100%
Product Development	Betsy McKenna	1/2 Time	50%
	Dedra Robertson	Full Time	50%
Graphical User Interface	Randy Singh	Full Time	50%
	Yoo Jun	Full Time	50%
QA/CM	Carl Audrey	Full Time	50%
Documentation	Patricia Lewis	Full Time	25%
	Patrick Kelly	Full Time	25%

Table 1-1: Project Personnel Matrix

Figure 1-1. Project Resources goes here.

SECTION 2.
SOFTWARE DEVELOPMENT MANAGEMENT

This section describes the management structure and the software development environment for the Airfields system. Configuration Management responsibilities are also defined. The software development environment for both hardware and software is described and any special security considerations are explained.

2.1   Program Configuration Management

Configuration Management as it applies to the Airfields system is the discipline for applying technical and administrative direction and surveillance that encompasses the following:

a.   Identify and document functional physical characteristics of all configuration items,

b.   Control all changes to those configuration items that form the baseline system, and

c.   Record and report on the status of all changes that are approved regarding processing and implementation.

The Airfields team will implement an internal configuration management system for control of all configuration documentation, physical media, and physical parts representing or comprising the product. For software, the system will address the evolving developmental configuration and support environments used to generate and test the product.

Detailed Configuration Management procedures are outlined in the Configuration Management Plan for the Joint Automated Message Editing System (JAMES) and Airfields Facilities File Information System (AIRFIELDS) dated January 1995.

2.2   Software Development Organization

The Airfields Designated Development Agency (DDA) is the Center for Software, Software Development Department (JEXA), General Applications Division (JEXAG). The Government flow of management control for Airfields' re-engineering is depicted in Figure 2-1, titled Organizational Structure.

Figure 2-1. Organizational Structure goes here.

2.3 Interface With Contractors

Software development for the re-engineering of the Airfields system will be accomplished by Government/in-house personnel only. The Ada Joint Program Office (AJPO) has contracted the mentoring aid of a software engineering firm, CACI.

2.4 Software Development Environment

The Airfields system will be re-engineered under the Unix environment on a Sun Solaris platform. Screen Machine, a Windows/Motif compliant Graphical User Interface (GUI) development tool which is compatible with Ada 95 and the Ada 95 compiler will be utilized.

2.5 Security

The Airfields software is being developed as an unclassified application using an unclassified subset of the current database. Upon completion of the development process, the system will be moved to the Global Command and Control System (GCCS) environment. The GCCS environment is a classified environment. At that time all retrievals, reports, etc. will be classified at "system high" (SECRET/ NOFORN). Security procedures will be accomplished as defined by the GCCS.

2.6 User Feedback Mechanisms

If analysis of a user incident report determines that there is an operational problem with Airfields, or the operational system does not fully comply with the system specifications, or there is an error in the documentation, a deficiency report (DR) will be generated. Airfields support personnel will work on the DR after it has been approved by the Program Management Office (PMO) and the Configuration Management Office (CMO).

2.7 Project Status Reporting

Airfields project personnel shall regularly inform the Ada Joint Program Office of the status of the project by submitting weekly Status Reports and attending bi-monthly Program Management Reviews (PMRs). Status information includes, but is not limited to, current position within the software development life cycle issues and concerns affecting the timely completion of the software, a

one-week look ahead, and a two-month look ahead.

SECTION 3.

    SOFTWARE DEVELOPMENT ACTIVITIES

3.1 <u>Release Concept</u>

The initial release of the re-engineered system will be moved to
and installed on the Global Command and Control environment.
Access to this newly-released system will be governed by security
procedures as outlined in the GCCS Integration Standards.
Following installation, any necessary user and/or operator training
may be accomplished.

3.2 <u>Definition Activity</u>

The Definition Activity is the first activity in the JIEO release
process. The objectives of this activity are:

    a.   to identify and define the requirements to be
          implemented in the release,

    b.   to ensure each requirement has a technical solution, and

    c.   to develop an initial release plan.

The result of this activity is the Functional/Requirements
Baseline.

3.2.1 <u>Procedures</u>

The Ada 95 team members will review the baseline Airfields system.
This review includes, but is not limited to, review of the
documentation, source code, and generated output. Procedures for
Release Activity will be accomplished as defined by the Global
Command and Control System.

3.2.2 <u>Reviews</u>

During re-engineering of the Airfields system, peer reviews, discussions, and output comparisons will be conducted/ accomplished periodically to ensure the functionality of the re-engineered system has not been compromised.

3.2.3 <u>Configuration Management</u>

In the initial stages of re-engineering, all baseline documentation was submitted to the Airfields configuration manager. During this activity the configuration manager, along with project personnel, began development/update of any required documentation.

3.3 <u>Specification Activity</u>

During the Specification Activity, the requirements identified in the Definition Activity are further analyzed and specified. The objectives of this activity are to:

    a.   Develop preliminary design for the Graphical User Interface,

    b.   Update and expand the Software Requirements Specification (SRS) document based on review of the current system,

    c.   Ensure the testability of each requirement, and

    d.   Determine and finalize method/procedure for installing/releasing system to GCCS environment.

3.3.1 <u>Procedures</u>

During the Specification Activity, preliminary designs shall be developed. The preliminary designs will allow Ada 95 team members to examine changes (if any) to user screens, report formats, graphical user interface development, etc. Additionally, any changes to required documentation will be accomplished during this phase.

3.3.2 <u>Reviews</u>

In accordance with MIL-STD-1521B, Preliminary Design Reviews (PDRs) shall be conducted by Ada 95 team members, Ada 95 management personnel, and the mentors assigned to the task. PDRs shall be accomplished during a regularly scheduled In-Progress Review (IPR).

3.3.3 Configuration Management

An Airfields Database Design Document (DBDD) shall be developed which will identify the database design for the system. Upon submission, the document shall be placed under formal control by the Configuration Manager.

3.4 Detailed Design Activity

The Detailed Design Activity provides the details on how each requirement will be implemented. The primary objectives of this activity are to (1) form the baseline for software implementation and (2) define the details of the technical solution for each requirement.

3.4.1 Procedures

A "top down structured" approach will be applied to the Airfields re-engineering effort. During this activity, detailed designs shall be developed and will identify any deviations from what was specified in the preliminary design specifications. An Airfields test plan shall also be developed during this activity.

3.4.2 Reviews

Critical design reviews, peer reviews, and code walk-throughs will be conducted as required during this activity.

3.4.3 Configuration Management

During this activity, the Airfields Software Test Plan (STP) shall be placed under formal configuration control by the Configuration Manager.

3.5 Implementation Activity

The following are activities which will occur during the Implementation Activity.

    a.   Translate GUI requirements into code

    b.   Develop preliminary design to satisfy current functionality of the system,

    c.   Translate the preliminary design for each requirement into code, and

    d.   perform developmental and system integration testing.

3.5.1 Procedures

The technical leader of the Airfields system will assign implementation tasks to Ada 95 team members based on the design of the Airfields and the experience of the team member.

3.5.2 Reviews

Peer reviews and walk-throughs will be conducted as required.

3.5.3 Configuration Management

During this activity, the Airfields code will be placed under formal configuration management control by the Configuration Manager.

3.5.4 Development Approach

Airfields will be re-engineered/developed using a Phased Approach. The Phases and the time allocated for completion of each phase follows:

    a.   Phase 1 allows for the incorporation of the Airfields One-Line Report into the application. This involves developing, integrating, and testing portions of the application code and the graphical user interface

relating to that phase. Completion of this phase is scheduled for 31 May 1995.

Following the completion of Phase 1, the GCCS environment development and testing activity will take place. This activity includes the following:

    (1) Acquiring the GCCS environment
    (2) Developing the installation package
    (3) Developing GCCS documentation
    (4) Testing in the GCCS environment
    (5) Correcting any environmental errors found
          under (4) above, and
    (6) Regression testing in the GCCS environment

Version 1.0 of the re-engineered system, which represents Phase 1 development only will be delivered on 31 May 1995.

b.   Phase 2 allows for the incorporation of the Airfields One-Page Summary reporting capability into the application. This involves developing, integrating, and testing portions of the application code and the graphical user interface relating to that phase. Completion of this phase is scheduled for 31 August 1995.

c.   Phase 3 allows for the incorporation of the Airfields Multi-Page reporting capability into the application. This involves developing, integrating, and testing portions of the application code and the graphical user interface relating to that phase. Completion of this phase is scheduled for 13 October 1995.

d.   Phase 4 allows for the incorporation of the Airfields Turnaround reporting capability into the application. This involves developing, integrating, and testing portions of the application code and the graphical user interface relating to that phase. Completion of this phase is scheduled for 22 September 1995.

e.   Phase 5 allows for the incorporation of the Airfields Selective Data retrieval reporting capability   into

the application. This involves developing, integrating, and testing portions of the application code and the graphical user interface relating to that phase. Completion of this phase is scheduled for 29 July 1995.

f.   Phase 6 allows for the development of the capability to perform range/radius calculations. This involves developing, integrating, and testing portions of the application code and the graphical user interface

relating to that phase. Completion of this phase is scheduled for 31 August 1995. Even though this phase is considered Phase 6, it is being done in conjunction with Phase 2.

Figure 3-1 represents a complete timeline of events.

3.5.5   Compilers

Airfields requires the use of the GNAT Ada 95 compilers. GNAT has the most mature Ada 95 compiler and it runs on the Sun Solaris 2.3.

3.5.6.   Graphical User Interface

Screen Machine will be chosen as the Graphical User Interface (GUI). Screen Machine is a Windows/Motif compliant GUI development tool that will be compatible with Ada 95 and Ada 95 compilers.

3.5.7   Database Management System

Airfields requires a database management system (DBMS) including Ada bindings. ORACLE Version 7 has been tentatively chosen as that DBMS.

3.5.8   Hardware Requirements

Airfields requires the use of eight Sun SparcStations to be used for development/re-engineering. Table 3-2 contains a matrix of hardware requirements.

PAGE 1 OF TIME LINE GOES HERE

PAGE 2 OF TIME LINE GOES HERE

PAGE 3 OF TIME LINE GOES HERE

PAGE 4 OF TIME LINE GOES HERE

PAGE 5 OF TIME LINE GOES HERE

PAGE 6 OF TIME LINE GOES HERE

Number	Platform	Operating System
1	Sun SparcStation 2	SunOS 4.1.3
8	80486 PC	MS-DOS 6.22
1	HP LaserJet 4	N/A

TABLE 3-2: Hardware Matrix

3.6   Acceptance Test Activity

The primary purpose of the Acceptance Test Activity is to verify
the correct and accurate implementation of all requirements. The
objectives of this activity are to:

a.   Execute the product in an environment as close to the
      operational environment as possible,

b.   Verify the capabilities to be fielded, and

c.   Demonstrate that the product is of acceptable quality
      for porting to the GCCS Common Operating Environment
      (COE).

Airfields requires the use of a test coverage analysis tool and a
test case generation tool.

3.6.1   Procedures

Side-by-side operation with the existing Airfields will provide a
"ground truth" environment that will determine if the re-engineered
system is giving the correct answer.

Test results shall be documented. Those errors which render the
system unusable, shall be immediately corrected.

3.6.2   Reviews

There are no formal reviews for this activity, however, management
and technical staff will review the results of the tests to
determine the priority for corrective action(s).

3.6.3   Configuration Management

All test documentation and results will be provided to the

Configuration Manager for control. This is in accordance with the Configuration Management Plan dated January 1995.

3.7 Release Activity

All release activity associated with the Airfields system will be accomplished by Global Command and Control System personnel.

3.7.1 Procedures

As determined by the Global Command and Control System.

3.7.2 Reviews

Not applicable.

3.7.3 Configuration Management

As determined by the Global Command and Control System.

3.8 Review Activity

The Review Activity is the last activity in the JIEO release process. The objectives of this activity are:

    a.   to review the success and/or failure of the release, and

    b.   to identify those areas of the development process needing improvement and to take corrective action.

This activity will be accomplished by the GCCS.

SECTION 4.   SOFTWARE CONFIGURATION MANAGEMENT PLAN

This section identifies the project's organization of its
Configuration Management (CM) element, addresses the overall change
control procedures and details how configuration control items are
identified and managed.

4.1   Organization

The Configuration Manager will be responsible for controlling the
existing software product and for reporting regularly on its
status.

4.2   Change Control

Airfields change control will be accomplished as determined by GCCS
standards.

4.3   Configuration Identification

Configuration identification will include the selection of
Configuration Items (CI's); the determination of the types of
configuration documentation required for each CI; and the issuance
of numbers and assigned identifiers for the CI's, their component
parts and associated configuration documentation, including
revision and version numbers where appropriate.

SECTION 5.   SOFTWARE QUALITY ASSURANCE PLAN

5.1 <u>Organization</u>

The Configuration Manager will also be responsible for Quality
Assurance which will consist of evaluating the software development
methodology as well as assessing compliance with software
engineering standards.

5.2 <u>Review Procedures</u>

The following types of activities will be reviewed as part of
overall quality assurance:

    a.   Review all documents for consistency, format and
          content, and compliance with appropriate standards,

    b.   Witness system software builds,

    c.   Witness system level tests including dry runs of
          acceptance/validation tests,

    d.   Review source code for consistency and compliance
          with coding standards, and

    e.   Support design reviews and code walk-throughs

5.3 <u>Testing Program</u>

Software testing is a critical element of software quality
assurance and represents the ultimate review of specification,
design, and coding. The Ada 95 Early Adoption projects shall
undergo a rigorous testing program. The following paragraphs
attempt to describe the program.

5.3.1 Testing Approach/Philosophy

Testing will be accomplished by selecting identical retrieval criteria on the WWMCCS COBOL system and the Sun Solaris Ada 95 version of Airfields and comparing the output received from the Sun Solaris against the output received from the WWMCCS.

An Airfields Test Plan which is under development will be used as a testing tool. Scenarios outlined in the test plan will be followed explicitly in order to ensure the system is conforming to functionality.

5.3.2 Testing Planning and Formal Documentation

Verification of functional performance will be accomplished based on the WWMCCS/COBOL version of Airfields. The functionality of the system should be identical to the original version.

Formal documentation consists of the following:

    a. Airfields Software Development Plan (SDP)

    b. Airfields System Requirements Specifications (SRS)

    c. Airfields/JAMES Configuration Management Plan CMP)

    d. Airfields Test Plan (STP)

    e. Airfields Test Description (STD)

    f. Airfields Test Report (STR)

    g. Airfields Software Center Operators Manual

    h. Airfields Database Design Document (DBDD)

    i. Airfields Software User Manual (SUM)

    j. Software Version Description (SVD)

    k. Software Installation Plan (SIP)

All documentation will be delivered with the system.

5.4 User Participation

User participation in the Ada 95 projects software quality assurance program is the key to the successful implementation of quality systems that meet the user community's requirements. The Ada 95 Configuration Manager will be the user community's representative during software quality assurance activities. His/her role in the process is to ensure the users' requirements are accurately and fully implemented.

5.5 Quality Objectives

Factors that affect software quality can be categorized in two broad groups: factors that can be directly measured (e.g., errors per thousand lines of code); and factors that can only be measured indirectly (e.g., usability or maintainability). In each case, measurement must occur. These software quality factors focus on two important aspects of a software product, its operational and maintenance characteristics.

The operational characteristics are comprised of the following:

a. Accuracy. The extent to which a system performs its functions with the required precision.

b. Correctness. The extent to which a system accurately and completely incorporates its functions and fulfills the users' mission objectives.

c. Efficiency. The amount of computing resources and code required by a system to perform its functions.

d. Integrity. The extent to which access to the system and/or data by unauthorized persons can be controlled.

e. Reliability. The extent to which a system can be expected to perform its intended functions.

f.   Usability. The effort required to learn, operate, prepare input, and interpret the output of the system.

The maintenance characteristics are comprised of the following:

a.   Flexibility. The effort required to modify an operational system.

b.   Interoperability. The effort required to couple one system to another.

c.   Maintainability. The effort required to locate and fix errors in the system.

d.   Portability. The effort required to re-host a system from one hardware and/or software platform to another.

e.   Reusability. The extent to which a system, or parts of a system, can be reused in another application.

f.   Testability. The effort required to test a system to ensure that it performs its intended functions.

SECTION 6.   SOFTWARE ENGINEERING

6.1   <u>Organization</u>

The following paragraphs describe the responsibilities of the Government in the software engineering of this project.

6.1.1 <u>Personnel Structure</u>

See Figure 1-1 for the structure of the software development team.

6.1.2   <u>Training</u>

Table 6-1 (below) illustrates the training needs of the Ada 95 team. As can be noted from the table, some of the courses have been competed.

Course	Provided By	To Attend	Status
Introduction to Ada	Keesler AFB	All	C o m p l e t e d 10/21/94
Advanced Ada	Keesler AFB	Technical Staff	C o m p l e t e d 11/11/94
Introduction to Ada 95	Alsys	All	C o m p l e t e d 12/2/94
Introduction to UNIX	DISA S.E.E	Technical Staff	C o m p l e t e d 2/17/95
Introduction to I-CASE	DISA S.E.E.	All	C o m p l e t e d 2/17/95
I n t r o d u c t i o n toOracle	TBD		TBD

TABLE 6-1:   Training Schedule

6.2   Standards

The following paragraphs describe the methodologies, coding
standards, and tools to be used in the development of the Ada 95
projects.

6.2.1   Methodologies

The design and coding of the Airfields will follow a top down
structured methodology.

6.2.2   Coding Standards .   Coding standards for Airfields
development will be as outlined in the following documents:

   a.   International Organization for Standardization,
        International Electrotechnical Committee, Information
        Technology Programming Languages Their Environments and
        System Software Interfaces, Ada 9X Quality and Style
        Guidelines for Professional Programmers (Draft Baseline
        Version) , SPC-94093-CMC, Version 00.01.00, dated
        February 1995

   b.   International Organization for Standardization,
        International Electrotechnical Committee, Information
        Technology Programming Languages Their Environments and
        System Software Interfaces, Ada 9X Reference Manual
        (Draft) , Version 5.0, dated 1 June 1994

   c.   International Organization for Standardization,
        International Electrotechnical Committee, Information
        Technology Programming Languages Their Environments and
        System Software Interfaces, Ada 9X Rationale (Draft),
        Version 5.0, dated 8 June 1994

6.2.3   Tools

The software depicted in Table 6-1 - Software Requirements Matrix
(below) is needed in order to meet the current requirements.

Tool	Vendor	Cost	Delivery Status
Ada 95 Compiler	GNAT	None	Installed
Data Binder	Microsoft	Unknown	TBD
Screen Machine	Objective Interface Systems	None – evaluation copy	1/3/95
DBMS	Oracle	TBD	TBD

Table 6-2: Software Requirements Matrix

6.3 Metrics Collection and Use

Measure the size, in bytes, of the Ada 95 version including and excluding the DBMS and GUI libraries and compare it with the COBOL version. The size of the executable code should be smaller than previous versions.

6.3.1 Increase Execution Speed

User interaction and processing should execute faster than previous versions of Airfields. Measure the time it takes to log into the system and select the criteria needed to build each type report and measure the time it takes to actually produce each type report. This will be done for each version of the system. The goal is to ensure that the Ada 95/re-engineered version builds and creates the reports in less time than the COBOL version.

APPENDIX A

TERMS AND ABBREVIATIONS

AFB   Air Force Base

AJPO   Ada Joint Program Office

CACI   A private Software Engineering Firm

CFSW   Center for Software

CI   Configuration Item

COBOL   Common Business Oriented Language

COE   Common Operating Environment

CM   Configuration Manager

CMO   Configuration Management Office

DBMS   Data Base Management System

DDA   Designated Development Agency

DISA   Defense Information Systems Agency

DMAAC   Defense Mapping Agency Aerospace Center

DoD   Department of Defense

DR   Deficiency Report

GCCS   Global Command and Control Systems

GUI   Graphical User Interface

IPR   In-Progress Review

JAMES   Joint Automated Message Editing System

JDSSC   Joint Data Systems Support Center

JIEO	Joint Interoperability & Engineering Organization
NOFORN	No Foreign [dissemination]
OS	Operating System
PDR	Preliminary Design Review
PMO	Program Management Office
QA	Quality Assurance
SDD	Software Design Document
SDP	Software Development Plan
SRS	Software Requirements Specifications
STD	Standard
STP	Software Test Plan
TBD	To Be Determined
WWMCCS	Worldwide Military Command and Control Systems